IN THE CLAIMS:

1. (Currently Amended) For use in a wide-issue pipelined processor, a mechanism for reducing pipeline stalls between conditional branches, comprising:

a mispredict program counter (PC) generator that generates a mispredict PC value for each conditional branch instruction in a pipeline of said processor; and

mispredict PC storage, coupled to said mispredict PC generator and including a mispredict PC queue and a number of staging registers, that stores said mispredict PC value at least until a resolution of said conditional branch instruction occurs and makes said mispredict PC value available to a PC of said processor if said resolution results in a mispredict condition, ~~a mispredict PC queue of said mispredict PC storage having more slots than said pipeline has stages~~ said mispredict PC queue having at least as many slots as said number of said staging registers.

2. (Original) The mechanism as recited in Claim 1 wherein said mispredict PC generator is associated with a static branch predictor of said processor.

3. (Original) The mechanism as recited in Claim 1 wherein said mispredict PC generator generates a branch prediction and said mispredict PC value in a single clock cycle.

4. (Original) The mechanism as recited in Claim 3 wherein said branch prediction is employed to prefetch instructions.

5. (Cancelled)

6. (Currently Amended) The mechanism as recited in Claim 1 wherein said mispredict PC value moves through said staging registers of said mispredict PC storage as said conditional branch instruction moves through stages in said pipeline.

7. (Original) The mechanism as recited in Claim 1 wherein said resolution occurs in an execution stage of said pipeline.

8. (Original) The mechanism as recited in Claim 1 wherein said processor is a digital signal processor.

9. (Currently Amended) For use in a wide-issue pipelined processor, a method of reducing pipeline stalls between conditional branches, comprising:

generating a mispredict program counter (PC) value for each conditional branch instruction in a pipeline of said processor; [[and]]

using a number of staging registers to align said mispredict PC value with said conditional branch instruction in said pipeline;

storing said mipredict PC value in a mispredict PC queue having at least as many slots as said number of said staging registers;

~~providing storage space for more mispredict PC values than said pipeline has stages; and~~

storing said mispredict PC value at least until a resolution of said conditional branch instruction occurs; and

making said mispredict PC value available to a PC of said processor if said resolution results in a mispredict condition.

10. (Original) The method as recited in Claim 9 wherein said mispredict PC generator is associated with a static branch predictor of said processor.

11. (Original) The method as recited in Claim 9 wherein said generating is carried out in a single clock cycle, said method further comprising generating a branch prediction in a single clock cycle.

12. (Original) The method as recited in Claim 11 further comprising employing said branch prediction to prefetch instructions.

13. (Cancelled)

14. (Currently Amended) The method as recited in Claim 9 further comprising moving said mispredict PC value through said staging registers in said mispredict PC storage as said conditional branch instruction moves through stages in said pipeline.

15. (Original) The method as recited in Claim 9 wherein said resolution occurs in an execution stage of said pipeline.

16. (Original) The method as recited in Claim 9 wherein said processor is a digital signal processor.

17. (Currently Amended) A digital signal processor, comprising:

a pipeline having stages capable of executing conditional branch instructions;

a wide-issue instruction issue unit;

a mispredict program counter (PC) generator that generates a mispredict PC value for each conditional branch instruction in said pipeline; and

mispredict PC storage, coupled to said mispredict PC generator and including a mispredict PC queue and a number of staging registers, that stores said mispredict PC value at least until a resolution of said conditional branch instruction occurs and makes said mispredict PC value available to a PC of said DSP if said resolution results in a mispredict condition, ~~a mispredict PC queue of said mispredict PC storage having more slots than said pipeline has stages~~ said mispredict queue having at least as many slots as said number of said staging registers.

18. (Original) The DSP as recited in Claim 17 wherein said mispredict PC generator is associated with a static branch predictor in said instruction issue unit.

19. (Original) The DSP as recited in Claim 17 wherein said mispredict PC generator generates a branch prediction and said mispredict PC value in a single clock cycle.

20. (Original) The DSP as recited in Claim 19 wherein said branch prediction is employed to prefetch instructions.

21. (Cancelled)

22. (Currently Amended) The DSP as recited in Claim 17 wherein said mispredict PC value moves through said staging registers in said mispredict PC storage as said conditional branch instruction moves through said stages.

23. (Original) The DSP as recited in Claim 17 wherein said resolution occurs in an execution stage of said pipeline.

24. (New) The mechanism as recited in Claim 1 wherein said mispredict PC queue of said mispredict PC storage has at least one more slot than said number of said staging registers.

25. (New) The method as recited in Claim 9 wherein said mispredict PC queue of said mispredict PC storage has at least one more slot than said number of said staging registers.

26. (New) The DSP as recited in Claim 17 wherein said mispredict PC queue of said mispredict PC storage has at least one more slot than said number of said staging registers.